BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

(12) UK Patent Application (19) GB (11) 2 366 025 (13) A

(43) Date of A Publication 27.02.2002

(21) Application No 0101741.7

(22) Date of Filing 23.01.2001

(30) Priority Data
(31) 12015227 (32) 25.01.2000 (33) JP

(51) INT CL[7]
G06F 12/10 // G06F 12/14

(52) UK CL (Edition T)
G4A AAP ANX

(56) Documents Cited
JP 100091531 A

(58) Field of Search
UK CL (Edition S) G4A AAP ANX
INT CL[7] G06F 1/00 9/32 12/10 12/14
ONLINE: WPI, EPODOC, PAJ

(71) Applicant(s)
Nintendo Co Ltd
(Incorporated in Japan)
11-1 Hokotate-cho, Kamitoba, Minami-ku, Kyoto, Japan

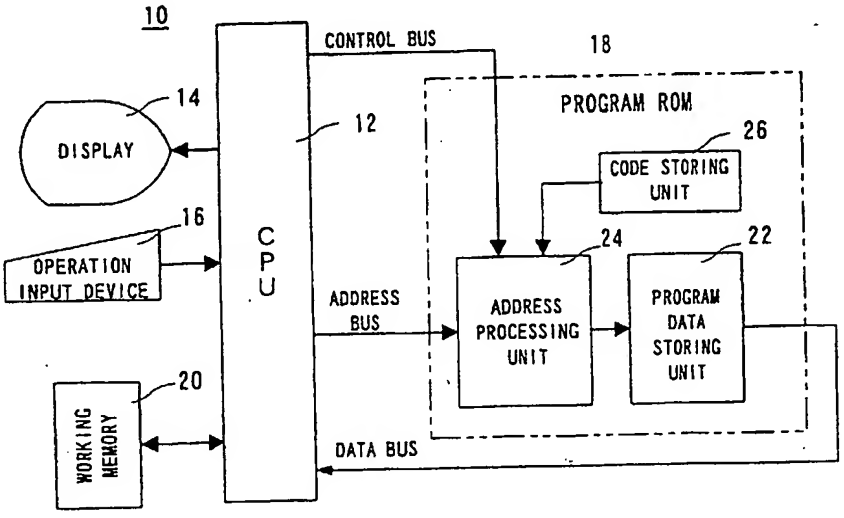
(72) Inventor(s)
Masaru Shimomura

(74) Agent and/or Address for Service
Stevens Hewlett & Perkins
Halton House, 20/23 Holborn, LONDON, EC1N 2JD, United Kingdom

(54) Abstract Title
Semiconductor storage device and program discrimination system

(57) A semiconductor storage device includes a program ROM, and a calculator of an address processing unit included in the program ROM calculates lower 8 bits A0 - A7 of first address data from an address bus of a CPU by using an address calculation code a0 - a7 of 8 bits being set in a code storing unit included in the same program ROM, and outputs a calculation result A0' - A7'. A0 - A7 is replaced with A0' - A7' by a selector, and therefore, second address data of 16bits in total of A0' - A7' + A8 - A15 is outputted from the address processing unit.

FIG. 1



GB 2 366 025 A

FIG. 1

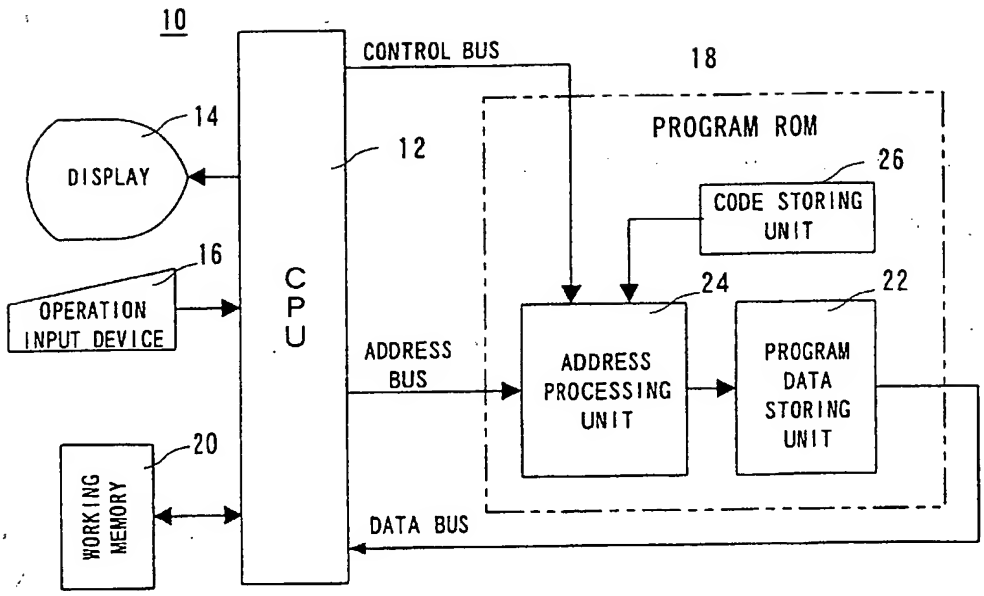

10
CONTROL BUS
18
PROGRAM ROM
14
DISPLAY
12
26
CODE STORING UNIT
16
OPERATION INPUT DEVICE
CPU
24
22
ADDRESS BUS
ADDRESS PROCESSING UNIT
PROGRAM DATA STORING UNIT
20
WORKING MEMORY
DATA BUS

FIG. 2

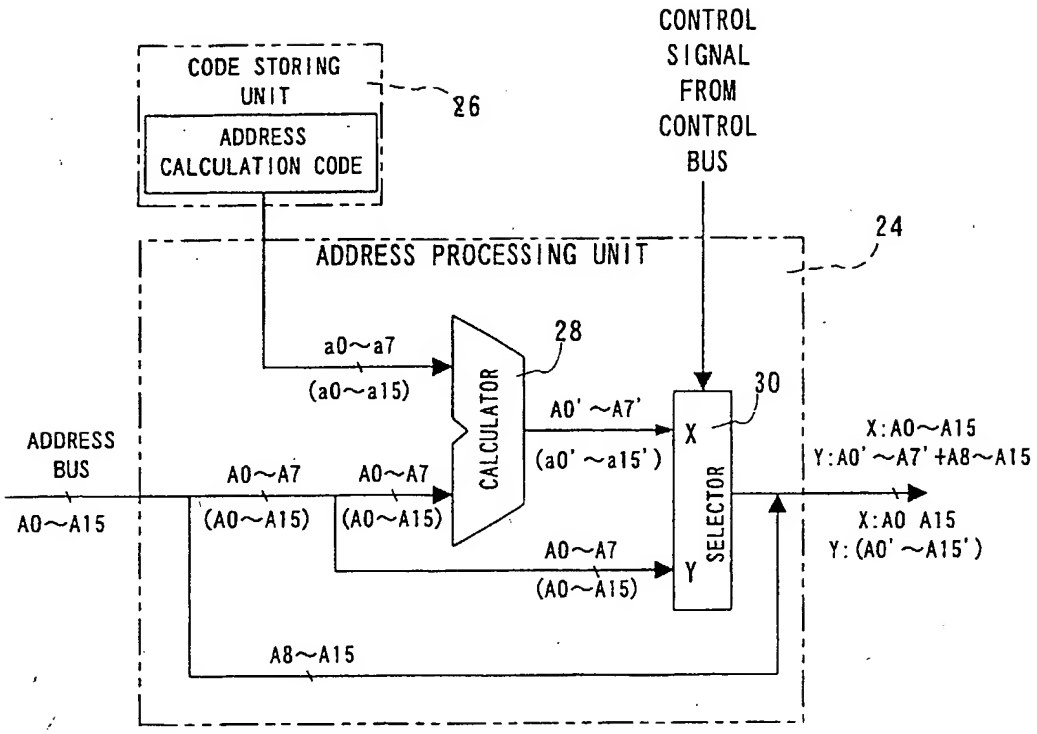

CONTROL SIGNAL FROM CONTROL BUS
CODE STORING UNIT
ADDRESS CALCULATION CODE
26
24
ADDRESS PROCESSING UNIT
28
a0~a7
(a0~a15)
CALCULATOR
A0'~A7'
(a0'~a15')
30
X
Y
SELECTOR
ADDRESS BUS
A0~A15
A0~A7
(A0~A15)
A0~A7
(A0~A15)
A0~A7
(A0~A15)
X:A0~A15
Y:A0'~A7'+A8~A15
X:A0 A15
Y:(A0'~A15')
A8~A15

FIG. 3

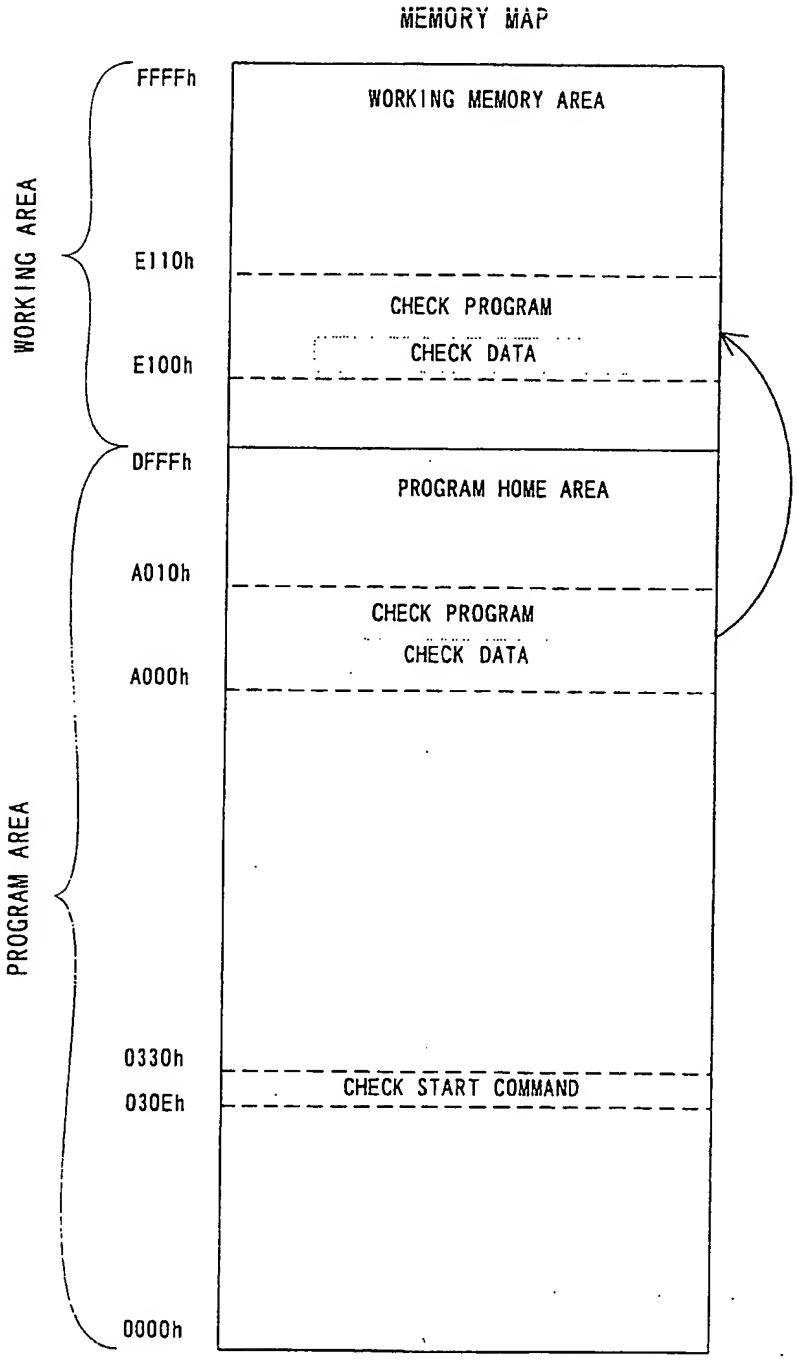

MEMORY MAP
FFFFh
WORKING MEMORY AREA
WORKING AREA
E110h
CHECK PROGRAM
CHECK DATA
E100h
DFFFh
PROGRAM HOME AREA
A010h
CHECK PROGRAM
CHECK DATA
A000h
PROGRAM AREA
0330h
CHECK START COMMAND
030Eh
0000h

FIG. 4

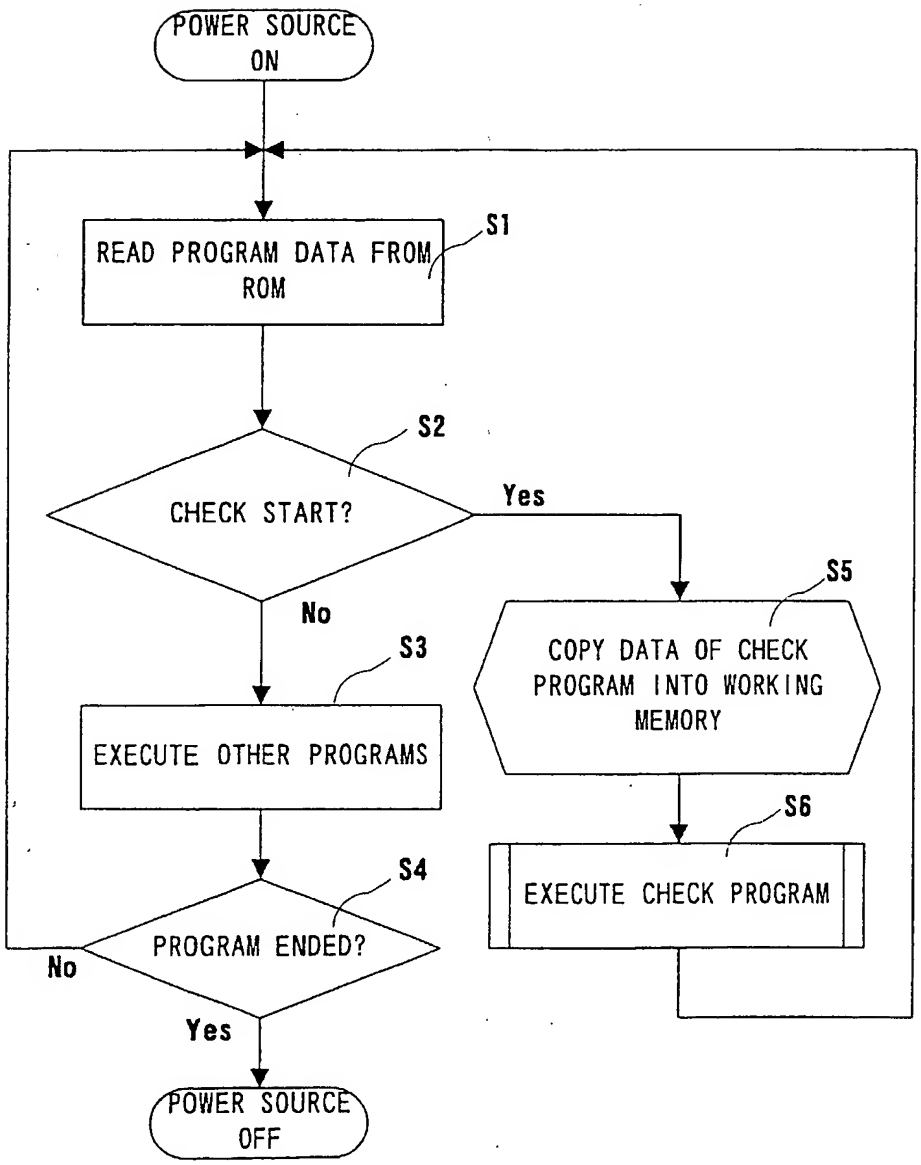

POWER SOURCE ON
READ PROGRAM DATA FROM ROM
S1
S2
CHECK START?
Yes
No
S3
EXECUTE OTHER PROGRAMS
S4
PROGRAM ENDED?
No
Yes
POWER SOURCE OFF
S5
COPY DATA OF CHECK PROGRAM INTO WORKING MEMORY
S6
EXECUTE CHECK PROGRAM

FIG. 5

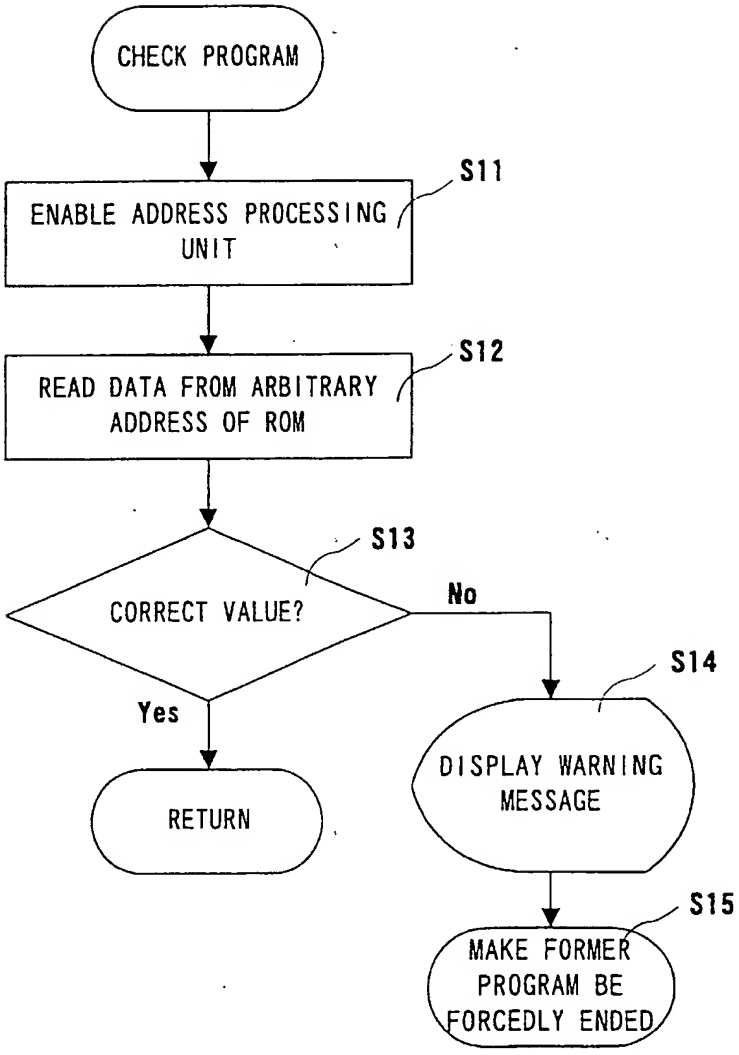

CHECK PROGRAM
S11
ENABLE ADDRESS PROCESSING UNIT
S12
READ DATA FROM ARBITRARY ADDRESS OF ROM
S13
CORRECT VALUE?
No
Yes
S14
DISPLAY WARNING MESSAGE
RETURN
S15
MAKE FORMER PROGRAM BE FORCEDLY ENDED

FIG. 6

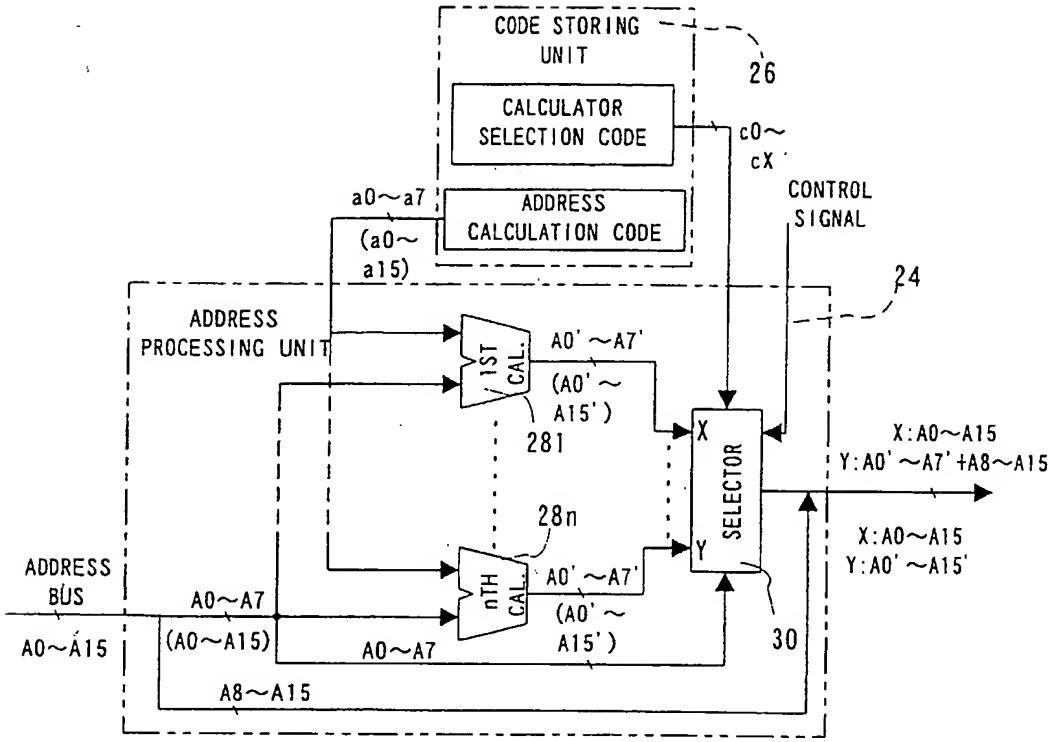

CODE STORING UNIT
26
CALCULATOR SELECTION CODE
c0~cX
ADDRESS CALCULATION CODE
a0~a7 (a0~a15)
CONTROL SIGNAL
24
ADDRESS PROCESSING UNIT
1ST CAL.
A0'~A7'
(A0'~A15')
281
28n
nTH CAL.
A0'~A7'
(A0'~A15')
X
Y
SELECTOR
30
X:A0~A15
Y:A0'~A7'+A8~A15
X:A0~A15
Y:A0'~A15'
ADDRESS BUS
A0~A15
A0~A7
(A0~A15)
A0~A7
A8~A15

FIG. 7

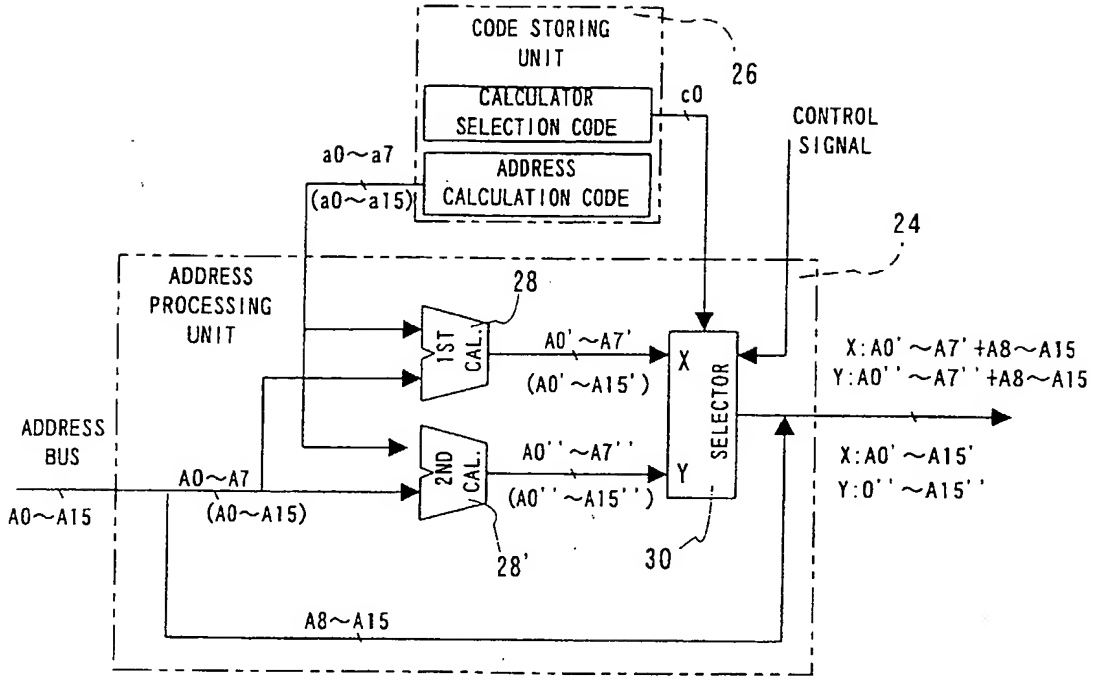
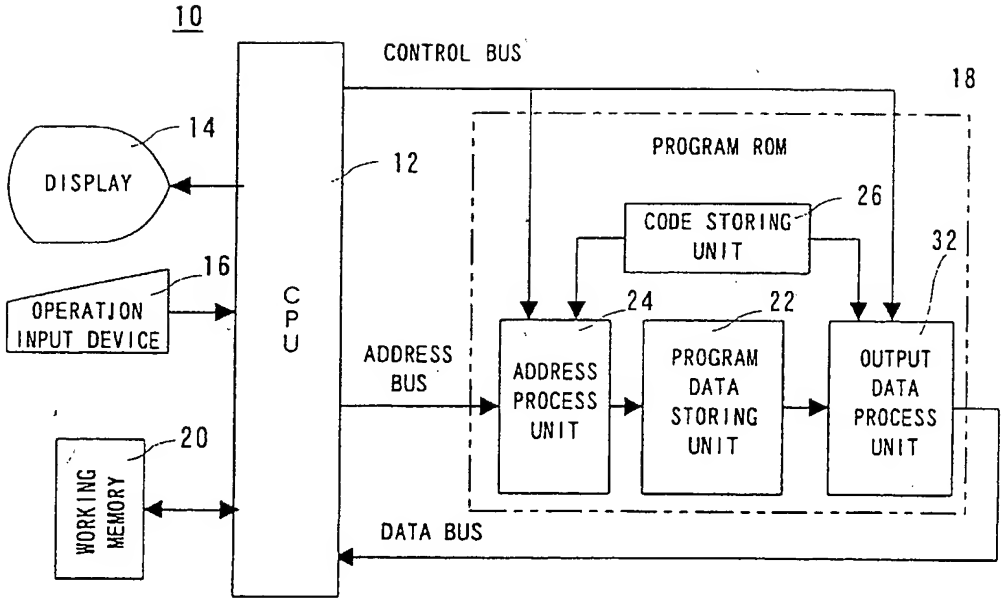

CODE STORING UNIT
26
CALCULATOR SELECTION CODE
c0
CONTROL SIGNAL
a0～a7
(a0～a15)
ADDRESS CALCULATION CODE
24
ADDRESS PROCESSING UNIT
28
1ST CAL.
A0'～A7'
(A0'～A15')
X
SELECTOR
Y
X:A0'～A7'+A8～A15
Y:A0''～A7''+A8～A15
ADDRESS BUS
A0～A7
(A0～A15)
2ND CAL.
A0''～A7''
(A0''～A15'')
X:A0'～A15'
Y:0''～A15''
A0～A15
28'
30
A8～A15

FIG. 8


10
CONTROL BUS
18
14
DISPLAY
12
PROGRAM ROM
26
CODE STORING UNIT
32
16
OPERATION INPUT DEVICE
CPU
24
22
ADDRESS BUS
ADDRESS PROCESS UNIT
PROGRAM DATA STORING UNIT
OUTPUT DATA PROCESS UNIT
20
WORKING MEMORY
DATA BUS

Semiconductor Storage Device and Program Discrimination System

The present invention relates to a semiconductor storage device and a program discrimination system using the same. More specially, the present invention relates to a semiconductor storage device such as a ROM which stores a program and etc. in a cartridge for game machine, for example, and a discrimination system of the program.

Description of the prior arts

For example, a cartridge for game machine includes a semiconductor storage device such as a ROM in which a game program is written, and is attached, while in use, to a game machine body. Especially, in a case of such a cartridge for game machine, a market is infested with imitations in each of which a game program is unlawfully copied.

Then, as disclosed in Japanese Patent Laying-open No. 2-31256 [G06F 12/14], the applicant proposed a system capable of preventing the game program and other programs from being unlawfully copied or the data from being changed by discriminating an authenticity and by inhibiting an access to a program memory in a case that a cartridge is not an authentic one.

In the above-described prior art, by using "Address Decoding Method", the authenticity is discriminated in accordance with program data actually read from the program memory, and therefore, it is possible to remove the imitations with a high reliability more than a certain degree.

However, in the above-described prior art, it is necessary to decode a number of addresses, and therefore, there was a problem to be further desolved that not only a chip

size becomes large because the number of gates becomes large but also if the addresses to be decoded have been analyzed, a protect effect against the unlawful utilization of the program is lost.

Therefore, a principal object of the invention is to provide a novel semiconductor storage device and a program discrimination system using the same.

Another object of the present invention is to provide a semiconductor storage device capable of making an analysis of a program more difficult and lasting a protection effect for a long time, and a program discrimination system using the same.

A semiconductor storage device according to the present invention is a semiconductor storage device in which program data to be executed by a central processing unit is fixedly stored and outputs the program data from addresses designated by address data, comprising: a program data storing means for fixedly storing the program data; a code generating means for generating an address calculation code; and an address processing means which receives first address data outputted from the central processing unit, wherein the address processing means outputs new second address data obtained by performing calculation on at least a portion of the first address data on the basis of the address calculation code at a time that a control signal exists and applies the second address data to the program data storing means, and applies the first address data to the program data storing means at a time that no control signal exists.

Specifically, the address processing means includes a calculating means capable of outputting the second address data by calculating at least the portion of the first address data by using the address calculation code, and a selector which outputs the second address data when the control signal is applied to the selector and outputs the first address

data when the control signal is not applied to the selector.

In such a case, the calculating means may include a plurality of calculators capable of performing different kinds of calculations, respectively and thus outputting different kinds of second address data, respectively on reception of the first address data and the address calculation code, and the code generating means further generates a calculator selection code by which any one of the plurality of calculators can be selected, and the selector may select the second address data associated with the calculator selected by the calculator selecting code, or the first address data.

In one aspect of the present invention, a semiconductor storage device comprises a program data storing means for fixedly storing program data; a code generating means for generating an address calculation code; and an address processing means which receives first address data outputted from a central processing unit, wherein the address processing means outputs new second address data obtained by performing a first calculation on at least a portion of the first address data based upon the address calculation code at a time that a control signal exists and applies the second address data to the program data storing means, and outputs new third address data obtained by performing a second calculation different from the first calculation on at least a portion of the first address data based upon the address calculation code at a time that no control signal exists and applies the third address data to the program data storing means.

In this case, one of the first calculation and the second calculation performed in the address processing means converts the first address data into address data for outputting program data that is to be regularly executed by the central processing unit.

A program discrimination system according to the present invention comprises a read-only program storage unit which fixedly stores program data and a central processing unit which executes the program data by reading the program data from the

program storage unit and discrimination whether or not the program is a regular program, and the program storage unit including a program data storing means for fixedly storing the program data, a code generating means for generating an address calculation code, and an address processing means for receiving first address data outputted from the central process unit, wherein the address processing means outputs new second address data obtained by performing a calculation on at least a portion of the first address data based upon the address calculation code when a control signal is applied from the central processing unit and applies the second address data to the program data storing means, and applies the first address data to the program data storing means when no control signal is applied, and the central processing unit including a control signal applying means for applying the control signal to the address processing means of the program storage unit, an address outputting means for outputting the first address data to apply the first address data to the address processing means just after the control signal is applied, and a determining means for determining an authenticity of the program storage unit by determining whether or not a predetermined relationship exists between the program data read from the program data storing means in accordance with the second address data and check data that is set in advance.

The central processing unit further includes a program terminating means for forcedly terminating the program when a determination result by the determining means indicates a false program storage unit.

In the semiconductor storage device, when the control signal which is outputted from the central processing unit in response to specific program data read from the program data storing means is applied to the address processing means, the address processing means outputs the second address data by performing a calculation on all bits or a part of bits of the first address data based upon the address calculation code from the

code generating means and applies the second address data to the program data storing means. Specifically, the calculating means of the address processing means calculates the first address data by using the address calculation code so as to output the second address data, and the selector outputs the second address data in response to the control signal. Therefore, from the program data storing means, in response to the control signal, the program data is read from an address that is designated by not the first address data outputted from the central processing unit but the second address data outputted from the address processing means.

In the central processing unit, it is determined by the determining means whether or not the program data read-out in accordance with the second address data is coincident with the check data being set in advance. If the both data are coincident with each other, the determining means determines that the semiconductor storage unit (program ROM) at that time is an authentic one. If the both data are not coincident with each other, the semiconductor storage device is a false one, and therefore, the program is forcedly terminated.

Accordingly, even if all the program data being stored in program data storing means are copied, in a case that the address processing means according to the present invention is not provided, the program storage unit is determined as an imitation according to a result of the program check, and therefore, it is impossible to execute the program.

In addition, when a plurality of calculators capable of executing different kinds of calculations, respectively are provided in the address processing means, the selector selects the second address data from a calculator selected by the calculation selection code outputted from the code generating means.

Furthermore, the second address data obtained by processing the first data through

the first calculation when the control signal exists or the third address data that is a result of the processing of the first address data through the second calculation when no control signal exists may be outputted. In this case, one of the first calculation and the second calculation in the address processing means converts the first address data into regular address data for outputting program data to be regularly executed by the central processing unit.

In accordance with the present invention, if only all the program data of the program data storing means are copied but all the elements such as the calculating means and the calculation code in the address processing means are not analyzed, the program data cannot be executed in different from the authentic product, and therefore, it is possible to effectively prevent the program from being unlawfully utilized.

Furthermore, in a case that the elements such as the kind of the calculation, the address calculation code in the calculating means and the calculator selection code by which the calculator is designated are changed for each product or version of product, even if all the elements of a given product could be analyzed, an analysis result cannot be applied to another product as it is, it is necessary for a person who intends to unlawfully use the program to newly analyze the elements on all such occasions. In view of a fact that a very long time and massive tools are needed for analysis, in accordance with the present invention, it is possible to substantively eliminate unlawful utilization of the program.

The above described objects and other objects, features, aspects and advantages of the present invention will become more apparent from the following detailed description of the present invention when taken in conjunction with the accompanying drawings.

Figure 1 is a block diagram showing one embodiment according to the present invention;

Figure 2 is a block diagram showing one example of an address processing unit of Figure 1 embodiment;

Figure 3 is an illustrative view showing a memory map of a CPU in Figure 1;

Figure 4 is a flowchart showing an operation of Figure 1 embodiment;

Figure 5 is a flowchart showing a check program in Figure 4 flowchart;

Figure 6 is a block diagram showing a modified example of the address processing unit;

Figure 7 is a block diagram showing another modified example of the address processing unit; and

Figure 8 is a block diagram showing another embodiment according to the present invention.

An information processing apparatus 10 of one embodiment according to the present invention and shown in Figure 1 includes a CPU 12 that is a central processing unit, and a display device 14 that is an output device and an operation input device 16 that is an input device are connected to the CPU 12. The information processing apparatus 10 constitutes a program discrimination system. When the information processing apparatus 10 is a game processor, the operation input device 16 is a game controller which includes a joystick, a cross key and various kinds of operating buttons, and the game processor, i.e. the CPU 12 processes a game program stored in a program ROM 18 in response to an operation of such the operation input device 16 so as to display a game image on the display device 14 that may be a television monitor or a liquid crystal display, for example.

In addition, a working memory 20 connected to the CPU 12 is constructed by a RAM and etc., for example, and utilized in a prosecution of the above-described program processing as necessary, and utilized for temporarily storing a check program and check data for a program authenticity discrimination described later.

The program ROM 18 which functions as a semiconductor storage device or a program storage unit is a read-only semiconductor memory such as PROM, flash ROM, EPROM, EEPROM and etc. which fixedly stores the program data. The program ROM 18 includes an address processing unit 24 and a code storing unit 26 in addition to a program data storing unit 22 which is a memory core equal to a conventional program ROM. Then, in the embodiment shown, the program data storing unit 22, the address processing unit 24 and the code storing unit 26 are provided on the same semiconductor chip. Furthermore, address data outputted from the CPU 12 through an address bus is processed by the address processing unit 24, and the address processing unit 24 applies processed address data to the program data storing unit 22. The program data storing unit 22 reads-out the program data in accordance with the address data processed by the address processing unit 24, and the program data is outputted to the CPU 12 through a data bus.

In addition, as described later, the address processing unit 24 processes the address data on the basis of a control signal applied through a control bus from the CPU 12 and a code (data) applied from the code storing unit 26.

A specific example of the address processing unit 24 is shown in Figure 2. As shown in Figure 2, the address processing unit 24 includes a calculator 28 which functions as a calculating means, and a selector 30. The calculator 28 is formed within the chip of the program ROM 18, and may be a multiplier, a divider, a subtractor, an adder, a shift register, or an other simple logic calculator (AND, NAND, OR, NOR,

EX-OR and etc., for example).

Upper 8 bits A8 - A15 of first address data A0 - A15 of 16 bits, for example, outputted from the CPU 12 are synthesized or coupled with output data A0' - A7' of a selector 30, and lower 8 bits A0 - A7 are applied to one input of the calculator 28 and the selector 30 as one input (X) thereof. To another input of the calculator 28, an address calculation code a0 - a7 of 8 bits being set in the code storing unit 26 which functions as a code generating means is applied. The calculator 28 executes any one of the above-described kinds of calculations on two inputs, and an output from the calculator 28 is applied to another input (Y) of the selector 30. In a case where the calculator 28 is AND, for example, the calculator 28 outputs a logical sum result A0' - A7' of the address calculation code a0 - a7 and the lower 8 bits A0 - A7 and applies the same to the above-described one input of the selector 30.

The control signal previously described, a write signal, for example is applied to the selector 30 from the CPU 12. Therefore, the selector 30 replaces the lower 8 bits A0 - A7 of the address data with the calculation result A0' - A7' in response to the control signal, i.e. the write signal and outputs replaced data. On the other hand, the upper 8 bits A8 - A15 of the address data A0 - A15 are coupled to the output of the selector 30, and resultingly, second address data of 16 bits in total of A0' - A7' + A 8 - A15 from the selector 30, i.e. the address processing unit 24 is outputted.

Since the second address data thus modified or processed by the address processing unit 24 is inputted to the program data storing unit 22 (Figure 1), in a step S12 shown in Figure 5 and described later, program data is read from an address of the program data storing unit 22 designated by not the first address data A0 - A15 but the second address data A0' - A7' + A8 - A15. In a case of a false program ROM which does not have the address processing unit 24, program data is read from an address shown by

the first address data A0 - A15. Therefore, the program data read from the program data storing unit 22 become different for each other. In a case of the former, the program data read-out is coincident with the check data being set in advance, but in a case of the latter, the program data is not coincident with the check data.

In addition, in Figure 2 embodiment, only the lower 8 bits of the first address data are calculated with the calculation code by the calculator 28, and the upper 8 bits are outputted as they are; however, all the 16 bits of the first address data may be calculated in the calculator 28. In such a case, as shown in Figure 2 with brackets, the all 16 bits of the address data A0 - A15 are applied to the one input of the calculator 28, and to the other input of the calculator 28, an address calculation code a0 - a15 of 16 bits from the code storing unit 26 is applied. Therefore, second address data A0' - A15' all the bits of which are modified or processed is outputted from the calculator 28, and the second address data A0' - A15' is applied to the one input (X) of the selector 30, and as the other input (Y) of the selector 30, the first address data A0 - A15 outputted from the CPU 12 is applied. Therefore, from the selector 30, when the control signal exists the second address data A0' - A15' is outputted, and when the control signal does not exist the first address data A0 - A15 is outputted. Therefore, in a step S13 in Figure 5, the propriety of the program data read from an address designated by the address data A0' - A15' is determined.

The CPU 12 in Figure 1 has a memory map shown in Figure 3, and a memory space "0000h - DFFFh", for example is a program area assigned to the program ROM 18, and a memory space of "E000h - FFFFh" is a working area assigned to the working memory 20.

The program data storing unit 22 of the program ROM 18 is a portion of the above-described program area, and the program area is further formed with a program stationing area or home area, a check program area and a check start command area. A

check program stored in the check program area can be implemented by flowcharts shown in Figure 4 and Figure 5 described later.

Furthermore, in the check program, one or plurality of check data is set. The check data is data for discriminating the authenticity of the program by comparing with data shown by a result of the execution of the check program. For example, in this embodiment shown, the authenticity of the program is checked only one time, and therefore, only one check data is set in the check program as illustrated in Figure 3. By setting the check start command for starting the check program which utilizes the single check data in a plurality of places of a main program (game program, for example), or by setting a plurality of check start commands for respectively starting a plurality of check programs utilizing a plurality of check data in a plurality of places, it is possible to make difficulty of the analysis of the program higher through the checking of the authenticity of the program at a plurality of times.

Furthermore, in a case that the check program is relatively small (a few to a few ten bytes, for example), it is preferably for making difficulty of analysis of the check program higher that the check program instead of the check start command is set in a plurality of places of the main program.

With referring Figure 4, when a power source (not shown) of the information processing apparatus 10 is turned-on, a step S1 is first executed by the CPU 12. In the step S1, the CPU 12 reads the program data stored in the program data storing unit 22 of the program ROM 18 (Figure 1). In a step S2, it is determined whether or not the read program data is specific program data which designates the execution of authenticity check of the program.

If "NO" is determined in the step S2, that is, if the program data is not the specific program data commanding the authenticity check, in a step S3, a process according to the

program data at that time is executed. In a step S4 after the step S3, the CPU 12 determines whether or not the program is ended, and if "YES" is determined, the process is terminated, but if "NO" is determined, the process returns the previous step S1. Thus, by repeating the steps S1 - S4, the program stored in the program data storing unit 22 is sequentially executed. In this state, the control signal (the write signal, for example) is not applied to the selector 30 of the address processing unit 22, and therefore, a state that the input X is selected is held in the selector 30, and accordingly, the first address data outputted from the CPU 12 is applied to the program storing unit 22.

If the command for executing the check program is inputted to the CPU 12 during a prosecution that respective commands included in the program area sequentially executed by repeating the steps S1 - S4, this is determined in the step S2, the process proceeds to a step S5. In a case that the command, i.e. the specific program data is set in an address "030Eh" as shown in Figure 3, the CPU 12 proceeds to the step S5 at a time that the address is incremented to the address "030Eh".

In the step S5, the CPU 12 reads-in the check program and check data both set in the program area shown in Figure 2, and writes or copies the both into the working memory 20. Therefore, a step S6 is executed in accordance with the check program and the check data copied in the working memory 20. However, the check program copied in the working memory 20 is eliminated after a completion of the check.

A subroutine of the step S6 is shown in Figure 5 in detail. In a step S11 of Figure 5, the CPU 12 first generates the control signal via the control bus to enable the address processing unit 24 included in the program ROM 18 shown in Figure 1. Specifically, the CPU 12 applies the write signal (control signal) to the selector 30 of the program ROM 18 through the control bus. The write signal is not normally applied to a ROM (read-only memory) during the execution of the program of the ROM, and therefore, by outputting

the write signal to the program ROM 18, the selector 30 of the address processing unit 24 is caused to select the input Y. The step S11 constitutes a control signal applying means.

In addition, instead of the above-described write signal is utilized as the control signal, a control signal may be applied to the selector 30 when the CPU 12 executes a specific address of the program data storing unit 22 of the program ROM 18. In such a case, although not shown, an address comparator which sequentially compares the address data outputted via the address bus from the CPU 12 with the specific address may be provided, and if it is detected that the both addresses are coincident with each other by the address comparator, the control signal is applied to the selector 30.

In a succeeding step S12, the CPU 12 outputs the address data (the first address data) for reading the program data storing unit 22 of the program ROM 18. More specifically, just after the step S11 wherein the control is outputted, in the step S12 which functions as an address input means, the first address data is inputted to the address processing unit 24 as a ROM address through the address bus from the CPU 12, but the address processing unit 24 processes the first address data as described in Figure 2 embodiment, and applies a processed result address (second address data) to the program ROM 18.

Then, in a step S13, it is determined whether or not the program data read from the program data storing unit 22 in accordance with the second address data processed by the address processing unit 24 is a true value. That is, the step S13 constitutes a determining means which determines whether or not the check data read in the working memory 20 and the program data are coincident with each other by comparing the both data. It is expected that the program data read from the address which is processed in accordance with a predetermined rule in the address processing unit 24 is coincident with the check data being set in advance. Therefore, in this case, "YES" is determined. However, in a

case that a false program ROM which is not provided with the address processing unit 24, or a false program ROM in which a unit equal to the address processing unit 24 exists but the address processing is not conformed to the predetermined rule, "NO" is determined in the step S13.

Then, in a case that the authentic or truth program ROM by determining as "YES" in the step S13, the process returns to a normal program processing. On the other hand, in a case of an imitation program ROM by determining as "NO", in a next step S14, the CPU 12 displays a warning message such as "This program ROM (cartridge) is an imitation, and thus, not used in this machine", for example on the display device 14 (Figure 1), and forcedly terminates the program in a step S15. Therefore, in a case of the imitation, it becomes impossible to continuously process of the program no longer. That is, the steps S14 and S15 are equal to a program forcedly terminating means.

In addition, in the above description, in the step S13 that is the determining means, it is compared whether or not the program data read-out in accordance with the second address data and the check data set in advance are coincident with each other, but the both may be not coincident with each other, and the both may have a predetermined relationship being set in advance. An arbitrary predetermined relationship, for example, a relationship that one is larger or smaller by a predetermined number in comparison with the other, or a relationship that the both becomes coincident with each other when a predetermined calculation is applied to one (and/or the other), or a relationship that absolute values of the both are equal to each other, or the like may be set.

In Figure 2 embodiment, if proper tools are utilized, the program data can be read from the program data storing unit 22 in a completion form, and copied; however, even if only the program data is restored from the program data storing unit 22, it is impossible to execute the restored program in different from the authentic product.

More specifically, even if only the program data is restored, in a case that the address processing unit 24 and the code storing unit 26 are not provided in a memory of a person who intends to unlawfully utilize the program data of the program ROM 18, when the program data designating the program check is read-out, the above-described address modification process cannot be executed, and therefore, the first address data A0 - A15 being not modified is inputted to the program data storing unit 22. Therefore, even if such the check command exists, the program data is read from an address designated by the first address data A0 - A15. In this case, the program data is different from the program data read from an address designated by the second address data A0' - A7' + A8 - A15 obtained in Figure 2 embodiment. Therefore, incoincidence is determined in comparing with the check data being set in advance in the step S13, and then, if the discrimination result of the incoincidence is obtained, "NO" is determined in the step S13, and therefore, the program is forcedly terminated.

In order to avoid such the forcedly termination of the program, it is necessary to analyze the specific program data which commands the check start in the step S2 in Figure 4 and the program step (address) thereof, and to further analyze all the elements such as a kind of the calculation in the calculator 28 and the address calculation code, and the check data set in the check program (Figure 2) and etc. In a case that the program ROM 18 is formed by a masked ROM, for example, it is very difficult to analyze such the elements, and therefore, a large-scale facility and a very long time are needed for the analysis. On the other hand, in a case that the elements such as the kind of the calculation in the calculator and the address calculation code are changed for each kind of the program ROM, game title in a case of the game cartridge, or a version of the program, even if all the elements of a given program ROM are analyzed, an analysis result cannot be applied to other program ROMs, and therefore, it is necessary for a person who intends

to unlawfully utilize the program to newly analyze all the elements at each time. Therefore, according to the embodiment shown, in view of the time and cost necessary for the analysis, it is possible to substantively avoid the unlawful utilization of the program ROM.

Figure 6 is a modified example of Figure 2 embodiment, and in Figure 6 embodiment, the calculating means of the address processing unit 24 includes a plurality of (n) calculators 281 - 28n. The lower 8 bits A0 - A7 of the first address data from the address bus are inputted to respective one inputs of the respective calculators 281- 28n, and respective other inputs commonly receive the address calculation code a0 - a7 of 8 bits outputted from the code storing unit 26. Furthermore, a calculator selection code c0 - cX being set in advance in the code storing unit 26 is applied to the selector 30 together with the control signal. The number of the bits of the calculator selection code c0 - cX may be set according to the number of the calculators (n), and in a case that four (4) calculators 28 exist, the number of the bits is 2, and in a case of eight (8) calculators, 3 bits may be used. Then, the calculator selection code c0 - cX is set to select any one of the plurality of calculators 281 - 28n.

In Figure 6 embodiment, as similar to the calculator 28 of Figure 2 embodiment, the respective calculators 281- 28n calculate or modify the address, and therefore, a specific description of such the address calculation may be omitted; however, in Figure 6 embodiment, it is set that the respective calculators 281 - 28n execute different kinds of calculations, respectively. Therefore, if different calculators can be selected by the calculator selection code c0 - cX for each kind of program ROM (game title) or a version of the program, the difficulty of the analysis of the program described in Figure 2 is further increased. Therefore, the unlawful utilization is made to be more difficult.

There is a further advantage in Figure 6 embodiment. More specifically, in a case

that only one calculator is formed in one chip as shown in Figure 2 and the calculator is changed according to the kind of product or the version of the program, if the program ROM is the masked ROM, for example, it is necessary to change a printing mask at every time that the calculator is changed. In contrast, if the plurality of calculators 281 - 28n are incorporated as shown in Figure 6, it is possible to select the calculator, that is, the kind of the calculation only by changing the calculator selection code. On the other hand, the calculator selection code can be set in the same printing step of the address calculation code, and therefore, in Figure 6 embodiment, it is possible to manufacture the program ROM with a low cost in a case that the calculator is changed.

In Figure 6 embodiment, the selector 30 selects the input X when the control signal is applied thereto, and selects the input Y when no control signal is applied, and therefore, the first address data or the second address data is outputted in accordance with absence or presence of the control signal. Furthermore, a case that all the bits of the address data A0 - A15 are modified and processed is shown with brackets in Figure 6. However, an operation of such a case can be easily understood in view of a previous description for Figure 2 embodiment, and thus, is omitted here.

In addition, in the above-described embodiments, when the control signal exists the address data (second address data) obtained by calculating a portion or a whole portion of the address data by the calculator 28 (281 - 28n) is outputted from the address processing unit 24, and when the control signal does not exist, the address data outputted from the CPU 12 (the first address data) is selected and outputted from the address processing unit 24. However, when no control signal exists, third address data obtained by a second calculation different from the calculation of the second address data on the first address data may be outputted.

Such an embodiment is shown in Figure 7. In Figure 7 embodiment, a second

calculator 28' is provided in the address processing unit 24 in addition to the calculator 28 and the selector 30 similar to those of Figure 2 embodiment. Then, to respective one inputs of the calculators 28 and 28', the lower 8 bits A0 - A7 of the address data A0 - A15 (first address data) from the address bus of the CPU 12 are applied as they are. To respective other inputs of the calculators 28 and 28', the address calculation code a0 - a7 of 8 bits being set in the code storing unit 26 are commonly applied. The calculators 28 and 28' are selected in a manner that the calculators 28 and 28' can execute a first kind of calculation and a second kind of calculation different from each other, respectively.

The calculator 28 outputs a first calculation result A0' - A7' of the address calculation code a0 - a7 and the lower 8 bits A0 - A7, and applies the same to the one input X of the selector 30. The calculator 28' outputs a second calculation result A0" - A7" of the address calculation code a0 - a7 and the lower 8 bits A0 - A7 to apply the same to the other input Y of the selector 30. One of these inputs A0' - A7' and A0" - A7" from the calculator 28 and 28' is selected by the selector 30. To the selector 30, the control signal, i.e. the write signal, for example is applied from the CPU 12 and the calculator selection code c0 - cX stored in the code storing unit 26 is applied. Therefore, the selector 30 selects the output from the calculator 28, i.e. the input X when the control signal exists, thereby to output the address data A0' - A7' of the lower 8 bits processed by the calculator 28. The selector 30 selects the output of the calculator 28', i.e. the input Y when the control signal is not applied, thereby to output the address data A0" - A7" of the lower 8 bits processed by the calculator 28'.

On the other hand, the upper 8 bits A8 - A15 of the address data A0 - A15 are coupled to the output of the selector 30, and resultingly, when the control signal is applied, from the selector 30, that is, the address processing unit 24, the address data (second address data) of 16 bits in total of A0' - A7' + A8 - A15 is outputted, and when the

control signal is not applied, from the selector 30, the address data (the third address data) of 16 bits in total of A0" - A7" + A8 - A15 is outputted.

In addition, in Figure 7 embodiment, as shown in Figure 7 with brackets, all the bits of the address data A0 - A15 may be calculated by the calculator 28 and the calculator 28',

In Figure 7 embodiment, the first address data A0 - A15 from the CPU 12 is not the address data by which the program data that the CPU 12 intends to normally execute can be read-out. That is, the first address data is merely false address data. Then, one of the first calculator 28 and the second calculator 28' changes or converts the first address data A0 - A15 such that the program data that the CPU 12 intends to normally execute can be correctly read-out. That is, one of the second address data and the third address data is correct address data.

In Figure 7 embodiment, in a case that the program is intended to be unlawfully utilized, it is necessary to analyze not only the first calculator 28 but also the second calculator 28', and therefore, the difficulty of the analysis is further increased.

The above-described embodiments are provided with the address processing unit 24 in the program ROM 18 to process or modify the address to be applied to the program data storing unit 22; however, it is possible to process or modify outputted program data as well as the address data by applying the same idea to the program data read from the program data storing unit 22.

Such an embodiment is shown in Figure 8. More specifically, in Figure 8 embodiment, the program ROM 18 includes the address processing unit 24 and the code storing unit 26 shown in Figure 1 embodiment in addition to the program data storing unit 22, and is further provided with an output data processing unit 32 which receives the program data outputted from the program data storing unit 22 and performs a

predetermined processing on the program data. Specifically, the output data processing unit 32 performs an operation similar to that of the above-described address processing unit 24 to process or modify the program data read from the program data storing u. In this case, the control signal from the CPU 12 is applied to the address processing unit 24 and the output data processing unit 32, and the both of the units 24 and the 32 are enabled in the step S11 in Figure 5.

In addition, in any one of the embodiments, the code storing unit 26 formed on the same chip of the program ROM 18 as the code generating means; however, the code generating means may be a means capable of applying an arbitrary code (data) to the calculating means, and therefore, the code storing unit 26 may be replaced with structure that arbitrary data is applied by a DIP switch and etc. or an external flash ROM stor: arbitrary data in a manner that the data can be rewritten from the outside.

Furthermore, in the above-described embodiments, in a case of the false program, the execution of the program is forcedly terminated at once; however, the program may be forcedly terminated after a predetermined time, or in a case of a game, such the forcedly termination may be replaced with other processing such as change of parameters of characters or elimination backup data of the game or returning the game to an initial state.

Although the present invention has been described and illustrated in detail, it is clearly understood that the same is by way of illustration and example only and is not to be taken by way of limitation, the spirit and scope of the present invention being limited only by the terms of the appended claims.

CLAIMS

1. A semiconductor storage device in which program data to be executed by a central processing unit is fixedly stored and outputs the program data from addresses designated by address data, comprising:

a program data storing means for fixedly storing the program data;

a code generating means for generating an address calculation code; and

an address processing means which receives first address data outputted from the central processing unit, wherein

the address processing means outputs new second address data obtained by performing calculation on at least a portion of the first address data on the basis of the address calculation code at a time that a control signal exists and applies the second address data to the program data storing means, and applies the first address data to the program data storing means at a time that no control signal exists.

2. A semiconductor storage device according to claim 1, wherein the address processing means includes a calculating means for calculating at least a portion of the first address data by using the address calculation code to output the second address data, and a selector which outputs the second address data at a time that the control signal is applied thereto and the first address data at a time that the control signal is not applied thereto.

3. A semiconductor storage device according to claim 2, wherein the calculating means includes a plurality of calculators respective of which can execute different kinds of calculations by receiving the first address data and the address calculation code and output different second address data,

the code generating means further generates a calculator selection code for selecting any one of the plurality of calculators, and

the selector selects the second address data associated with the calculator selected by the calculator selection code or the first address data.

4. A semiconductor storage device according to any one of claims 1 to 3, wherein the program data storing means includes specific program data which commands to output the control signal during execution of the program by the central processing unit.

5. A semiconductor storage device according to any one of claims 1 to 4, wherein the program data storing means, the code generating means and the address processing means are formed on a single semiconductor chip.

6. A semiconductor storage device according to any one of claim 1 to 5, wherein the program data storing means and the code generating means are constructed by a flash memory.

7. A semiconductor storage device in which program data to be executed by a central processing unit is fixedly stored and outputs the program data from addresses designated by address data, comprising:

a program data storing means for fixedly storing the program data;

a code generating means for generating an address calculation code; and

an address processing means which receives first address data outputted from the central processing unit, wherein

the address processing means outputs new second address data obtained by performing a first calculation on at least a portion of the first address data on the basis of the address calculation code at a time that a control signal exists and applies the second address data to the program data storing means, and applies third address data obtained by performing a second calculation different from the first calculation on at least a portion of the first address data on the basis of the address calculation code to the program data storing means at a time that no control signal exists.

8. A semiconductor storage device according to claim 7, wherein one of first and second calculations of the address processing means converts the first address data into address data for outputting program data to be normally executed by the central processing unit.

9. A program discrimination system, comprising:

a read-only program storage unit which fixedly stores program data, and

a central processing unit which executes the program data by reading the program data from the program storage unit and discrimination whether or not the program is a regular program, wherein

the program storage unit includes

a program data storing means for fixedly storing the program data,

a code generating means for generating an address calculation code, and

an address processing means for receiving first address data outputted from the central process unit, wherein

the address processing means outputs new second address data obtained by performing a calculation on at least a portion of the first address data based upon the address calculation code when a control signal is applied from the central processing unit and applies the second address data to the program data storing means, and applies the first address data to the program data storing means when no control signal is applied, and

the central processing unit includes

a control signal applying means for applying the control signal to the address processing means of the program storage unit,

an address outputting means for outputting the first address data to apply the first address data to the address processing means just after the control signal is applied, and

a determining means for determining an authenticity of the program storage unit by determining whether or not a predetermined relationship exists between the program data read from the program data storing means in accordance with the second address data and check data that is set in advance.

10. A program discrimination system according to claim 9, wherein the control signal applying means includes a specific program being set in advance in the program storing means, and the control signal is applied at a timing that the specific program is read-out.

11. A program discrimination system according to claim 9, wherein the central processing unit further includes a program terminating means for forcedly terminating the program when a determination result by the determining means indicates a false program storage unit.

12. An address control method for a semiconductor storage device including a program ROM which fixedly stores program data to be executed by a central processing unit, comprising steps of:

(a) generating an address calculation code;

(b) outputting new second address data obtained by applying a calculation to at least a portion of first address data outputted from the central processing unit on the basis of the address calculation code at a time that a control signal exists so as to apply the second address data to the program data storing means; and

(c) applying the first address data to the program data storing means at a time that the control signal does not exist.

13. An address control method for a semiconductor storage device including a program ROM which fixedly stores program data to be executed by a central processing unit, comprising steps of:

(a) generating an address calculation code;

(b) outputting new second address data obtained by applying a first calculation to at least a portion of first address data outputted from the central processing unit on the basis of the address calculation code at a time that a control signal exists so as to apply the second address data to the program data storing means; and

(c) outputting new third address data obtained by applying a second calculation different from the first calculation to at least a portion of the first address data on the basis of the address calculation code at a time that the control signal does not exist so as to apply the third address data to the program data storing means.

14. A program discrimination method in an information processing apparatus being provided with a read-only program storage device which fixedly stores program data and a central processing unit which executes a program by reading the program data from the program storage device, comprising steps of:

(a) generating an address calculation code;

(b) outputting a control signal;

(c) outputting a first address from the central processing unit just after the step (b);

(d) outputting new second address data obtained by applying a calculation to at least a portion of first address data outputted from the central processing unit on the basis of the address calculation code at a time that a control signal exists so as to apply the second address data to the program data storing means;

(e) applying the first address data to the program data storing means at a time that the control signal does not; and

(f) determining whether or not the program data read from the program data storing means according to the second address data and check data being setting in advance have a predetermined relationship in the central processing unit.

15. A semiconductor storage device substantially as hereinbefore described with reference to and as shown in the accompanying drawings.

16. A program discrimination systems substantially as hereinbefore described with reference to and as shown in the accompanying drawings.

The Patent Office

INVESTOR IN PEOPLE

Application No:	GB 0101741.7	**Examiner:**	Pierre Oliviere
Claims searched:	1-16	**Date of search:**	6 December 2001

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.S): G4A (AAP, ANX)

Int Cl (Ed.7): G06F (1/00, 9/32, 12/10, 12/14)

Other: Online: WPI, EPODOC, PAJ.

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
A	JP 100091531 A (TOKYO SHIBAURA ELECTRIC CO)	

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.